A survey of codes and algorithms used in NERSC material science allocations

Lin-Wang Wang
NERSC System Architecture Team
Lawrence Berkeley National Laboratory

We have carried out a survey of codes and algorithms used on NERSC computers within the science category of material science. This is part of the effort to track the usage of different algorithms in NERSC community. This survey is based on the data provided in the ERCAP application of FY06. To figure out the usage of each code in one account, we have multiplied the total high performance computer (HPC) time allocation (MPP hours) of this account with the percentage usage of this code as estimated by the users in the ERCAP application. This is not the actual usage time, but should be a good estimation of it, and it represents the intention of the users. In a few cases (~5) where the estimated usages are not provided, rough values are used based on the numbers in similar projects and the author's experience in the related field. Different groups might have used slightly different names for a same code. This has been corrected manually by the author. The statistics is done based on the original allocation. The additional allocation after Bassi is online has roughly doubled the total allocation.

We have the following observations based on our survey:

(1) There are 65 accounts under the BES/material science category, they account for 20% of the 324 total NERSC accounts. The total HPC allocation for these accounts is 4.1 Mhours before the Bassi was online, 8.8Mhours after the Bassi was online. These account for 13% of the total NERSC allocations (66.7Mhours after the Bassi was online). This percentage is smaller than a few years ago, partly because the increase of some other categories, like fusion, and partly because the creations of special programs like SciDAC and INCITE.
(2) There are in total 62 major codes used in these 65 accounts, thus in average about one code per account (or say user group). However, since the same code can be used by different accounts, in average one code is used by 2.15 user groups. Except the VASP code, which is used by 23 groups, the majority of the codes is used by less than 5 groups, and many of the codes are used only by one group. This is a very diverse community, with many groups using their own codes.
(3) The different codes can be classified into 6 categories based on their physics and the corresponding algorithms. They are: DFT (density functional theory); beyond DFT (GW+BSE); QMC (quantum Monte Carlo); CMD (classical molecular dynamics); CMC (classical Monte Carlo); and other PDE (Partial differential equation). Their corresponding HPC time usages are: 74.0%, 6.9%, 6.7%, 6.4%, 3.1%, and 2.9%, as represented in Fig.1. Thus, the majority of the time is spent on the DFT method, owing to the current success of that method in ab initio material science simulation. Within the DFT method, based on their different numerical

approaches, they can be divided into: Plane Wave DFT, Green's function DFT, localized basis and orbital DFT, Maffin Tin sphere type DFT, and real space grid DFT (as listed in Table.II). The most popular (both in terms of number of codes and the HPC hours) one is the plane wave DFT. There are 12 codes for Planewave DFT, and account for 1.6 Mhours (before Bassi) (see Table.II). A more detailed explanation of these codes and algorithms will be given in the last paragraph.

All the 62 codes are listed in Table.I, along with the number of user groups, and the estimated HPC hours. A short description is also included for each code. This table is also presented as a plot in Fig.1. The data in Table.I has been regrouped in Table.II, divided into different types of codes, e.g, planewave LDA, localized orbital LDA, etc. Finally, the results in Table.II are summarized in Fig.2 and Fig.3.

As for mathematical algorithms and libraries, the information we can get from the ERCAP application is very limited. It might not be a reliable source to gauge which library is used and by what percentage of time. We do notice that many group indicate the usages of ESSL, fftw, lapack and scalapack. However, there is no information for which subroutines are used in these libraries, and by how much. But in general, we do feel that the above libraries are extremely important. At this point, we also do not know, for a typical material science code, how much time is spent on library routines, and how much time is spent on the rest of the code (e.g., the Fortran part written by users). Different extremes exist. For example, for a typical planewave DFT code, the majority of the time is spent on the user written Fortran code. But for a beyond DFT GW+BSE code, the majority of the time is spent on solving a dense linear algebra problem using, e.g, scalapack.

Finally, we like to provide a more detailed description of difference methods shown in Fig.3. This is most to help us to understand what are the relevant mathematical aspects and computer science issues. First, in the DFT (density functional theory) method, one needs to solve the single particle Schrodinger's equation (a second order partial differential equation). Typically 5-10% of the lowest eigen vectors are needed from this Schrodinger's equation (eigenstate equation). The number of eigen vectors is proportional to the number of electrons in the system. For example, for a thousand atom system, a few thousand eigen vectors are needed. This is a major difference to most engineering problems (e.g, fluid dynamics, climate simulation, combustion, where a small fixed number of time evolving fields are solved, and for Maxwell equation where a few electric magnetic eigen vector fields are solved). In DFT, the Schrodinger's equation (eigen state matrix) itself depends on the eigen vectors through the density function (thus the name of density functional theory). Thus it is a nonlinear problem. This nonlinear problem can be solved by selfconsistent iterations of the linearized eigenstate problem (Schrodinger's equation), or by direct nonlinear minimization. Currently, most large scale calculations are done using selfconsistent iterations. Numerically, what distinguish the different DFT methods and codes are the different basis sets used to describe the wavefunctions (the eigen vectors). Planewave DFT uses planewaves to describe the wavefunctions, while real space DFT uses a regular real space grid. Due to the sharp peak of the potential near the atomic nuclei, special cares are needed to choose different basis

set. Besides the planewave and real space grid, the other conventional basis sets include: atomic orbital basis set where the eigen vectors of the atomic Schrodinger's equation are used to describe the wavefunctions in a solid or molecule; Gaussian basis set which is more often used in quantum chemistry due to its analytical properties; Muffin-tin basis where a spherical hole is cast out near each nuclei and spherical Harmonics and Bessel functions are used to describe the wavefunction inside the hole; Augmented planewaves where spherical Bessel functions near the nuclei are connected with the planewaves in the interstitial regions and used as the basis set; and the wavelet basis sets. In terms of the methods to solve the eigenstate problem, both iterative scheme and direct eigensolvers have been used in different codes. In the planewave DFT, iterative method (e.g, conjugated gradient method) is often used. While in the atomic orbital, Gaussian, and Augmented planewave (FLAPW) methods direct dense solvers (scalapack) are often used. For real-space grid method, sparse matrix solver is used. For the iterative solver, the most time consuming steps are the matrix vector multiplication and vector-vector multiplication (for orthogonalization). For the planewave DFT, the FFT is one bottleneck for large processor calculations.

GW+BSE is one approach to calculate the excited states and optical spectrums. It requires large dense matrix. The most time consuming parts are to generate these matrix and diagonalize the matrix (for its eigen vectors). The diagonalization part is often done using dens eigen solver (scalapack). The dimension of the matrix is proportional to the square of the number of the electron in the system. Quantum Monte Carlo (MC) method uses stochastic random walk to carry out the multidimensional integral of the many body wavefunctions. Since it needs an assemble sum of different independent walkers, it is possible for embarrassing parallelization. Quantum MC is a very accurate method, but it suffers from statistical noises, thus it is difficult to be used for atomic dynamics (where the forces on the atoms are needed). For the classical molecular dynamics (MD), the parallelization is done in the step of force calculations. Since the classical force field formalism is local in nature (except the electrostatic force), efficient parallelization is possible as in the code of NAMD. Classical MC some time is used to replace the classical MD, thus it is more interested in the time evolving process, instead of an assemble sum (like in quantum MC). As a result, the parallelization is not so trivial. There are many recent developments for how to develop parallel schemes for classical MC (besides the possible approach for parallel evaluation of the total energy like in classical MD). Other PDE includes Maxwell equations (e.g, in photonic study), and time evolving differential equation for grain boundary and defect dynamics, etc.

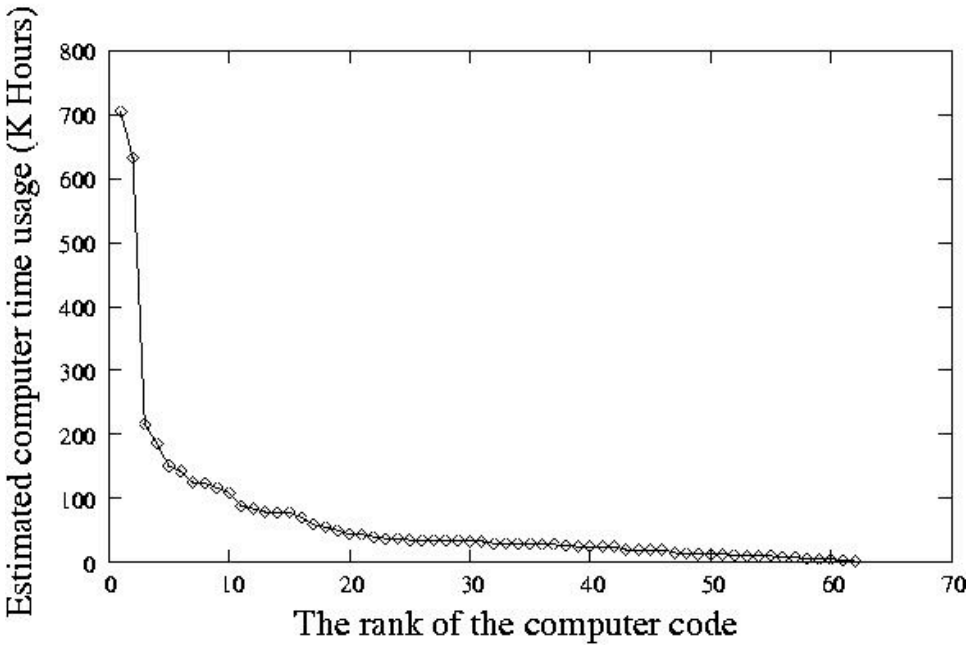

Fig.1, the computer time usage of different codes. Each symbol represents one code.

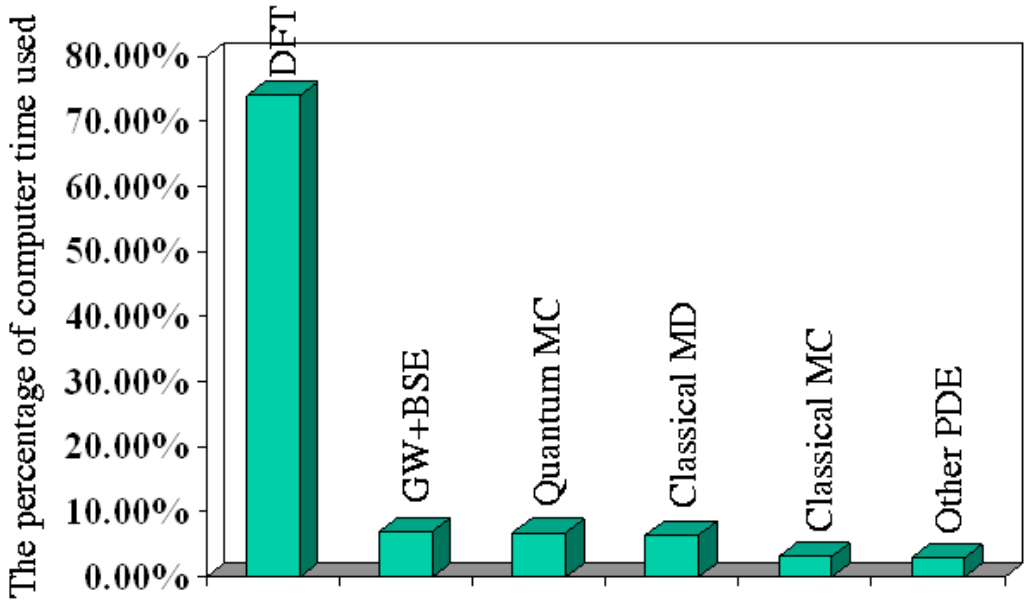

Fig.2: The percentage of computer time used for codes belong to different categories. The workload is dominated by Density Functional Theory (DFT) codes.

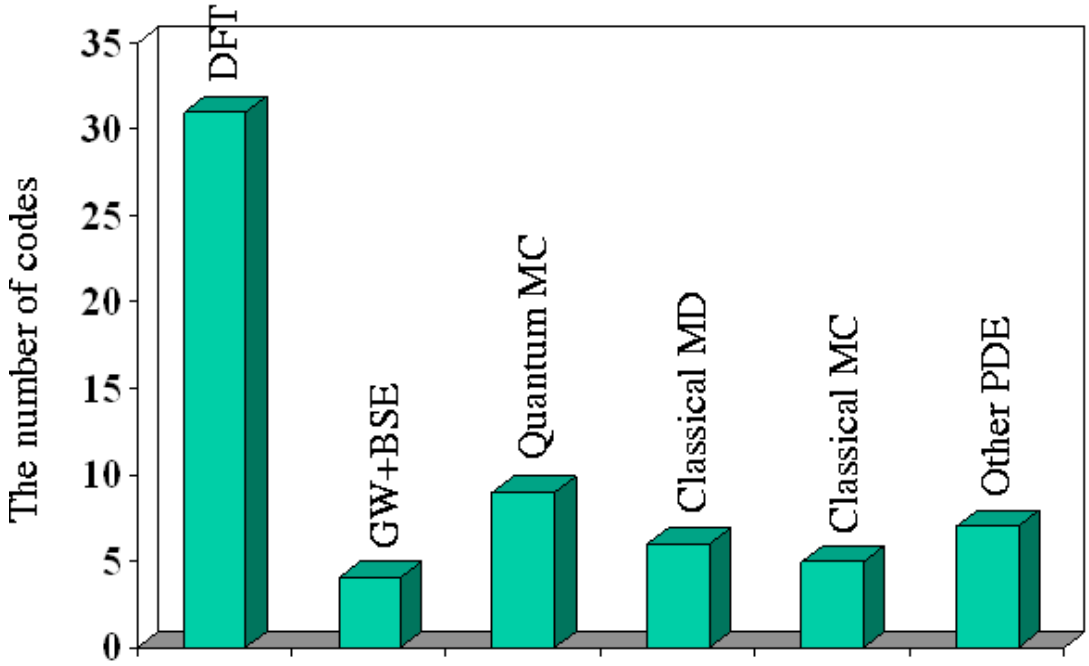

Fig.3: The number of codes belonging to different categories.

Table.I, The list of all the codes used in Material Science on NERSC machines. N_user is the number of groups (accounts) using that code. HPC(KH) is the estimated high performance computer time (MPP hours) used for each code measured in thousand hours (KH).

rank	N-user	HPC(KH)	Code name	Narrative
1	23	704.5	VASP	Vienna, planewave DFT
2	3	632.4	LSMS	ORNL, local mutiple scattering DFT
3	4	216.0	PEscan	LBNL, planewave Folded Spectrum Method
4	3	186.0	Scarlet	Berkeley, quantum transport, atomic basis
5	2	151.0	ALCMD	Classical MD
6	6	142.5	Paratec	Berkeley, planewave DFT
7	2	125.0	Qbox	LLNL, massively parallel planewave DFT
8	3	124.0	CMAT	NREL, Coulomb calc, and CI
9	6	116.5	PWSCF	Planewave DFT
10	2	109.0	FLAPW	NW Univ, FLAPW
11	4	89.00	GW	Berkeley, GW
12	2	84.0	PEtot	LBNL, planewave DFT
13	4	79.0	SIESTA	Spain, local basis/orbital DFT
14	3	78.75	TBMD	Tight-binding Molecular Dynamics
15	2	78.75	BO-LSD-MD	Planewave DFT MD
16	1	70.0	CASINO	Quantum MC
17	3	59.0	DLPOLY	Classical MD
18	4	55.5	NWchem	NWchem DFT, planewav and local basis
19	1	50.0	CHAMP	Quantum Monte Carlo
20	2	43.75	Multigrid	North Carolina, real space DFT
21	1	43.75	XqmmmX	QM/MM, PW for QM, Amber for MM
22	1	40.0	SSEqmc	Quamtum MC for lattice spin
23	3	37.5	Parsec	Minnesota, real space DFT
24	3	37.5	RGWBS	Minnesota, G-space TDLDA
25	1	35.0	LMTO-SIC	LMTO, DFT, Self-interaction correction
26	1	35.0	Psi-Mag	Classical spin-MC
27	1	35.0	SGF	Surface green's function
28	1	35.0	Moldy	Force field classical MC
29	1	35.0	OLCAO	LCAO TB like DFT
30	3	33.0	NAMD	Classical force field MD
31	3	33.0	BSE	Berkeley, Bethe-Salpeter Eq.
32	2	30.0	FDTDn	Genetic Algorithm, classical el-mag, energy
33	1	30.0	Flair	FLAPW LDA
34	1	30.0	CF-Monte-Carlo	Quatum Monte Carlo for model H
35	2	30.0	Abinit	Planewave DFT
36	1	28.0	Dmol3	Atomic orbital DFT
37	1	28.0	FLMTO	LMTO DFT
38	1	26.0	Polycrys	Plastic Deformation, grain boundary
39	1	25.0	LAMMPS	Particle method molecular dynamics
40	1	25.0	PWDFT	Planewave DFT
41	1	25.0	QMC-DCA	Quantum MC and dynamic cluster
42	1	25.0	SPF	Classical spin-phonon-Fermion MC
43	1	20.0	Qdot	Quantum MC for 2D electron
44	1	20.0	FEFF	Multi-scattering Green's func. electronic st
45	1	20.0	becmw	Ordinary Differential Equation.
46	1	20.0	Qmhubbard	Quantum Mechanics, hubbard model
47	1	15.0	Freepar$_1$x	Special algorithm Quantum MC
48	1	14.0	TransG98	Green's func. transport, Gaussian basis
49	1	12.5	AndyS	George Tech. Planewave DFT
50	1	12.5	CL-GCMD	Classical MD, complex liquid
51	1	12.5	Hollicita	2D vertices PDE
52	1	12.0	Mol-dyn	London Eq. MD, particle method
53	1	10.0	Moment	Maxwell Eq, FFT, photonics
54	1	10.0	TMM	Transfer matrix for Maxwell Eq, photonic
55	1	10.0	BEST	Planewave DFT
56	2	8.0	AMber	Classical force field MD
57	1	8.0	MC	Classical MC for vortices
58	1	6.0	ARPEsmpi	Multiple scattering photoemission
59	1	5.0	AFQMC	William Mary, Quantum Monte Carlo
60	1	5.0	WIEN2K	Vienna, FLAPW
61	1	3.75	Hartree	Real-space Hubbard Model, FH, LDA
62	1	2.0	Ginger	Classical particle
Total		4100.0		

Table.II, the computer codes in Table.I grouped into different categories.

rank	N_user	HPC(KH)	Code name	Narrative
Plane Wave DFT				N_code=12, HPC(total)=1588.5KH
1	23	704.5	VASP	Vienna, planewave DFT
3	4	216.0	PEscan	LBNL, planewave Folded Spectrum Method
6	6	142.5	Paratec	Berkeley, planewave DFT
7	2	125.0	Qbox	LLNL, massively parallel planewave DFT
12	2	84.0	PEtot	LBNL, planewave DFT
15	2	78.75	BO_LSD_MD	Planewave DFT MD
21	1	43.75	XqmmmX	QM/MM, PW for QM, Amber for MM
35	2	30.0	Abinit	Planewave DFT
40	1	25.0	PWDFT	Planewave DFT
49	1	12.5	AndyS	George Tech. Planewave DFT
55	1	10.0	BEST	Planewave DFT
Multiple Scattering (Green's Function) DFT				N_code=4, HPC(total)=693.4KH
2	3	632.4	LSMS	ORNL, local mutiple scattering DFT
44	1	20.0	FEFF	Multi-scattering Green's func. electronic st
27	1	35.0	SGF	Surface green's function
58	1	6.0	ARPEsmpi	Multiple scattering photoemission
Localized basis DFT and TB				N_code=7, HPC(total)=476.25KH
4	3	186.0	Scarlet	Berkeley, quantum transport, atomic basis
13	4	79.0	SIESTA	Spain, local basis/orbital DFT
14	3	78.75	TBMD	Tight-binding Molecular Dynamics
18	4	55.5	NWchem	NWchem DFT, planewav and local basis
29	1	35.0	OLCAO	LCAO TB like DFT
36	1	28.0	Dmol3	Atomic orbital DFT
48	1	14.0	TransG98	Green's func. transport, Gaussian basis
GW+BSE type (dense linear algebra)				N_code=4, HPC(total)=283.5KH
8	3	124.0	CMAT	NREL, Coulomb calc, and CI
11	4	89.00	GW	Berkeley, GW
24	3	37.5	RGWBS	Minnesota, G-space TDLDA
31	3	33.0	BSE	Berkeley, Bethe-Salpeter Eq.
Quantum Monte Carlo (wavefunction and spin)				N_code=9, HPC(total)=275.0KH
16	1	70.0	CASINO	Quantum MC
19	1	50.0	CHAMP	Quantum Monte Carlo
22	1	40.0	SSEqmc	Quamtum MC for lattice spin
34	1	30.0	CF_Monte_Carlo	Quatum Monte Carlo for model H
41	1	25.0	QMC_DCA	Quantum MC and dynamic cluster
43	1	20.0	Qdot	Quantum MC for 2D electron
46	1	20.0	Qmhubbard	Quantum Mechanics, hubbard model
47	1	15.0	Freepar.x	Special algorithm Quantum MC
59	1	5.0	AFQMC	William Mary, Quantum Monte Carlo
Classical force field molecular dynamics				N_code=6, HPC(total)=265.5KH
5	2	151.0	ALCMD	Classical MD
17	3	59.0	DLPOLY	Classical MD
30	3	33.0	NAMD	Classical force field MD
50	1	12.5	CL_GCMD	Classical MD, complex liquid
56	2	8.0	AMber 1	Classical force field MD
62	1	2.0	Ginger	Classical particle
Maffin Tin sphere type (including FLAPW) DFT				N_code=5, HPC(total)=207.0KH
10	2	109.0	FLAPW	NW Univ, FLAPW
25	1	35.0	LMTO_SIC	LMTO, DFT, Self-interaction correction
33	1	30.0	Flair	FLAPW LDA
37	1	28.0	FLMTO	LMTO DFT
60	1	5.0	WIEN2K	Vienna, FLAPW
Classical force field Monte Carlo				N_code=5, HPC(total)=128.0KH
26	1	35.0	Psi_Mag	Classical spin-MC
28	1	35.0	Moldy	Force field classical MC
39	1	25.0	LAMMPS	Particle method molecular dynamics
42	1	25.0	SPF	Classical spin-phonon-Fermion MC
57	1	8.0	MC	Classical MC for vortices

continue				
rank	N_user	HPC(KH)	Code name	Narrative
Other PDE (many real space grid)				N_code=7, HPC(total)=120.5KH
32	2	30.0	FDTDn	Genetic Algorithm, classical el-mag, energy
38	1	26.0	Polycrys	Plastic Deformation, grain boundary
45	1	20.0	becmw	Ordinary Differential Equation.
51	1	12.5	Hollicita	2D vertices PDE
52	1	12.0	Mol_dyn	London Eq. MD, particle method
53	1	10.0	Moment	Maxwell Eq, FFT, photonics
54	1	10.0	TMM	Transfer matrix for Maxwell Eq, photonic
Real space grid DFT				N_code=3, HPC(total)=85.0KH
20	2	43.75	Multigrid	North Carolina, real space DFT
23	3	37.5	Parsec	Minnesota, real space DFT
61	1	3.75	Hartree	Real-space Hubbard Model, HF, LDA
Total		4100.0		